Supporting Text

Description of the algorithm.

We write the alignment score as the sum of a *link score*

$$S^{\ell}(\mathbf{a},\mathbf{b},\pi)=\sum_{i,i'}^{N_A} s^{\ell}(a_{ii'},b_{\pi(i)\pi(i')}) \quad [15]$$

and a *node score*

$$S^{n}(\mathbf{\Theta},\pi)=\mu n_0+(\lambda_n+\mu)n_1+(\lambda_n'+\mu)n_2\ . \quad [16]$$

n_0 is the number of aligned gene pairs where neither node has an orthologous partner, n_1 is the number of orthologous aligned node pairs, and n_2 is the number aligned node pairs which are not orthologous to each other, but where either partner has an ortholog other than the alignment partner (see Fig 1 c). The generalization of the nodescore to more general measures such as **8** is straightforward.

Mapping alignments to permutations. It turns out to be useful to place N_B additional nodes, termed dummy-nodes, in graph A (with dummy entries in the adjacency matrix which do not contribute to the score), and to add N_A dummy nodes to graph B. Formally, the two graphs now have the same number of nodes $N=N_A+N_B$. A one-to-one alignment π can thus be considered as a permutation $\pi: j=\pi(i)$, where nodes without an alignment partner are formally aligned with a dummy node.

For the minimal scoring function $s^{\ell}(a,b)=ab$, the link score **15** then amounts to the trace of a product of the adjacency matrices $\mathbf{a},\mathbf{b}$ and the permutation matrix π, $S^{\ell}(\mathbf{a},\mathbf{b},\pi)=\mathrm{Tr}\left(\mathbf{a}\pi\mathbf{b}\pi^{T}\right)$. Finding the maximum of this trace over π is an NP-hard problem known as the *quadratic assignment problem* (1). A heuristic solution of this problem proceeds iteratively through a series of permutations $\ldots,\pi_{n-1},\pi_n,\pi_{n+1},\ldots$, where successive permutations are solutions of a *linear assignment problem* $\pi_{\mathbf{n+1}}=\mathrm{argmax}_{\pi}\mathrm{Tr}\left(\mathbf{a}\pi\mathbf{b}\pi_{\mathbf{n}}{}^{T}\right)$ (2). The linear assignment problem can be solved in polynomial time (3), and algorithms are available which take $O(N^3)$ steps per iteration. We utilize this strategy, first to treat general scoring functions $s^{\ell}(a,b)$, and then to treat the full score **13**.

Link score. The link score **15** can be written as the trace of a matrix as follows

$$S^{\ell}(\mathbf{a},\mathbf{b},\pi)=\sum_{i,k=1}^{N} s^{\ell}(a_{ik},b_{\pi(i)\pi(k)})=\mathrm{Tr}(\pi\mathbf{M}^{\pi})\ ,$$

where the $N\times N$ matrix $\mathbf{M}^{\pi}$ has elements given by

$$M_{ij}^{\pi}=\sum_{k=1}^{N} s^{\ell}(a_{jk},b_{i\pi(k)})\ .$$

We consider a series of permutations $\ldots,\pi_{n-1},\pi_n,\pi_{n+1},\ldots$, with

$$\pi_{\mathbf{n+1}}=\mathrm{argmax}_{\pi}\mathrm{Tr}(\pi\mathbf{M}_{\mathbf{n}}^{\pi})\ . \quad [17]$$

We observe that for symmetric adjacency matrices, the expression $\mathrm{Tr}(\pi_{\mathbf{n}}\mathbf{M}^{\pi_{\mathbf{n-1}}})$ monotonously increases from one iteration to the next

$$\begin{aligned}&\mathrm{Tr}(\pi_{\mathbf{n+1}}\mathbf{M}^{\pi_{\mathbf{n}}}) \quad [18]\\ &=\sum_{i,j} s^{\ell}(a_{ij},b_{\pi_{n+1}(i)\pi_n(j)})\geq\sum_{i,j} s^{\ell}(a_{ij},b_{\pi_{n-1}(i)\pi_n(j)})\\ &=\sum_{i,j} s^{\ell}(a_{ji},b_{\pi_n(j)\pi_{n-1}(i)})=\mathrm{Tr}(\pi_{\mathbf{n}}\mathbf{M}^{\pi_{\mathbf{n-1}}})\ ,\end{aligned}$$

where the $\geq$ sign holds for any π_{n-1} by construction. Thus $\mathrm{Tr}(\pi_{\mathbf{n}}\mathbf{M}^{\pi_{\mathbf{n-1}}})$ increases from one iteration to the next, until it converges to a (possibly local) maximum. We will use a random noise term to prevent the algorithm from getting stuck in a local score maximum. The same approach is applicable to directed graphs as well, see below.

Node score. We now turn to the node score **16**, which can also be written as the trace of a matrix

$$S^{n}(\mathbf{\Theta},\pi)=\mathrm{Tr}(\pi\mathbf{R})\ ,$$

where the entries of the matrix $\mathbf{R}$ are defined as

$$R_{ji}=\begin{cases}\lambda_n+\mu & \text{if node } i \text{ in } A \text{ is orthologous to } j \text{ in } B.\\ \mu & \text{if neither } i \text{ nor } j \text{ has an orthologous partner.}\\ \lambda_n'+\mu & \text{if either } i \text{ or } j \text{ has an orthologous partner, but it is not } j \text{ or } i\text{, respectively.}\\ 0 & \text{if either } i \text{ or } j \text{ is a dummy node.}\end{cases}$$

The full alignment score **13** can now easily be written as

$$S(\mathbf{a},\mathbf{b},\mathbf{\Theta},\pi,m) = \mathrm{Tr}\left(\pi\left(\mathbf{M}^{\mathbf{P}}+\mathbf{R}\right)\right) \ . \quad [19]$$

Maximizing this score over the alignment denoted by π thus represents a mixture of a generalized quadratic assignment problem (the link score) and a linear assignment problem (the node score).

Steps of the algorithm. Based on the iterative procedure **17** and **18** our heuristic to solve the full problem of maximizing the score **19** over the alignments proceeds as follows

1. Begin with $\pi' = \mathbb{1}$ and set β to β_{start}.
2. Find the permutation π maximizing $\mathrm{Tr}\left(\pi\left(\mathbf{M}^{\pi'}+\mathbf{R}+\mathbf{\Xi}/\beta\right)\right)$, where $\mathbf{\Xi}$ is an $N \times N$ matrix with entries chosen at random at each step from a normal distribution with mean zero and variance one.
3. Set π' equal to π and increase β by β_{step}
4. Repeat from 2. a predetermined number of times.

Throughout we increased the noise parameter β from 0.5 to 10 in 15 steps. It turned out that the final results depend only very little on the details of this annealing schedule. Both for the alignment of human-human networks, as well as for the alignment of human-mouse co-expression data, the algorithm converged after about 5 iterations.

We used the linear assignment routine by Jonker and Volgenant (3), which has a running time of $O(N^3)$ for each step. Note that the construction of the matrix $\mathbf{M}^{\pi}$ also requires of order N^3 steps. The algorithm by Jonker and Volgenant is based on finding the shortest path from unassigned rows to unassigned columns, using the corresponding entry of the assignment matrix as a path length. The actual running time for a single iteration for $N = 3200$ (allowing for a sufficient number of dummy nodes) was about 90 minutes on a Apple PowerPC G5 at 2 GHz.

Maximum score alignments and parametric optimization. At fixed values of the scoring parameters, the algorithm described above produces high-scoring alignments. The properties of these alignments depend strongly on the values of the scoring parameters. The link score function is inferred from the alignment specified by orthologs, see main text. For the parameters of the node score **16**, this approach is not feasible: the node score parameters quantify, for instance, the degree of deviation of the alignment from these orthology relations. Instead, we infer the values of the node score parameters from maximum likelihood, by maximizing the likelihood **12** over the node score parameters. For the binary orthology relation, the ensembles **8** can be written as

$$\begin{aligned} p_0^n(\theta) &= e^{\zeta_0\theta}/Z_0 \\ q_1^n(\theta) &= e^{(\lambda_n+\zeta_0)\theta}/Z_1 \\ q_2^n(\theta) &= e^{(\lambda'_n+\zeta_0)\theta}/Z_2 \ , \end{aligned}$$

with $Z_0 = 1 + e^{\zeta_0}$, $Z_1 = 1 + e^{\lambda_n+\zeta_0}$, etc. Given alignment π and the matrix of orthology relations $\mathbf{\Theta}$, the maximum-likelihood values $\zeta_0^*, \lambda_n^*, \lambda_n'^*$ can easily be determined by maximizing **12** with respect to the parameters $\zeta_0, \lambda_n, \lambda'_n$. One obtains

$$\begin{aligned} h_0 &= (N_A-p)(N_B-p)e^{\zeta_0^*}/(1+e^{\zeta_0^*}) \\ n_1 &= pe^{\lambda_n+\zeta_0}/(1+e^{\lambda_n+\zeta_0}) \\ n_2 &= p(N_A+N_B-1-p)e^{\lambda'_n+\zeta_0}/(1+e^{\lambda'_n+\zeta_0}) \ , \end{aligned}$$

where h_0 is the number of orthologous node pairs where neither node has an alignment partner, n_1 is the number of orthologous aligned node pairs, and n_2 is the number aligned node pairs which are not orthologous to each other, but where either partner has an ortholog other than the alignment partner. p denotes the number of aligned node pairs. The value of the chemical potential μ follows immediately with $\mu = \log\left(\frac{1+e^{\zeta_0}}{1+e^{\zeta_0+\lambda}}\right) + (N_A+N_B-1-p)\log\left(\frac{1+e^{\zeta_0}}{1+e^{\zeta_0+\lambda'}}\right)$.

A convenient way to determine the alignment with maximal score and the optimal scoring parameters is to exploit the iterative nature of the alignment algorithm. Following each iteration of the algorithm we determine the maximum-likelihood values $\zeta_0^*, \lambda^*, \lambda^{'*}$, and use the resulting scoring parameters in the next iteration.

Directed graphs. One can treat directed graphs in this fashion as well. Here we focus on binary graphs, the extension to weighted graphs is straightforward. A directed binary graph A can be encoded in a symmetric matrix $\mathbf{a}'$ with

$$\begin{array}{ll} a'_{ij} = 1 & \text{if } a_{ij} = 1 \text{ and } i < j \text{ or if } a_{ji} = 1 \text{ and} \\ & i > j \\ a'_{ij} = -1 & \text{if } a_{ji} = 1 \text{ and } i < j \text{ or if } a_{ij} = 1 \text{ and} \\ & i > j \end{array} .$$

Graph B is encoded analogously at each step with

$$\begin{array}{ll} b'_{ij} = 1 & \text{if } b_{ij} = 1 \text{ and } \pi^{-1}(i) < \pi^{-1}(j) \text{ or if} \\ & b_{ji} = 1 \text{ and } \pi^{-1}(i) > \pi^{-1}(j) \\ b'_{ij} = -1 & \text{if } b_{ji} = 1 \text{ and } \pi^{-1}(i) < \pi^{-1}(j) \text{ or if} \\ & b_{ij} = 1 \text{ and } \pi^{-1}(i) > \pi^{-1}(j) \end{array} .$$

The iterative step **18** now works as before, provided the alignments at consecutive steps are sufficiently similar to each other.

Co-expression networks

The expression data were taken from the experiments of Su et al. (4), which give expression levels of genes across a wide range of tissues both in humans and mice. We selected subsets of genes of each organism to construct co-expression networks. The genes were chosen to have a low standard deviation of the expression patterns (housekeeping genes), or have a high correlation with one of those genes. Subsets chosen according to the opposite criterion (high variation of the expression profiles) were also tested, the proof of principle described in the section *Results* gave very similar results.

We selected all genes with a standard deviation of the expression pattern below 0.4 (in *H. sapiens*) and 0.25 (in *M. musculus*), resulting in approximately 850 genes in each organism. Then all genes with a Spearman correlation coefficient of more than 0.68 (in *H. sapiens*) and 0.56 (in *M. musculus*) with one or more of these housekeeping genes were selected, as well as their orthologs in either organism. The mouse-human orthologs were taken from the *Ensembl Genome Browser* (5) accessed using the R-project (6) and the BiomaRt package (7,8). This resulted in $N_A = 2165$ genes of *H. sapiens* and $N_B = 2165$ genes of *M. musculus* with 2052 putative orthologous node pairs between them. Some nodes have several putative orthologs; in both networks there are 2040 nodes with one or more putative ortholog.

Then the Spearman correlation coefficients were calculated for each pair of genes in the two gene sets. For two sets of data $\{x_i\}$ and $\{y_i\}$ both containing N values, the Spearman correlation coefficient ρ is defined as the correlation coefficient of the ranks $\{r_i^x\}$ and $\{r_i^y\}$ of $\{x_i\}$ and $\{y_i\}$,

$$\rho = \frac{\sum_i (r_i^x - \bar{r^x})(r_i^y - \bar{r^y})}{\sqrt{\sum_i (r_i^x - \bar{r^x})^2 \sum_j (r_j^y - \bar{r^y})^2}} ,$$

where the overbar denotes the average $\bar{r^x} = (1/N) \sum_i r_i^x$. Spearman's rank correlation coefficient is a non-parametric measure of correlation particularly suited for the analysis of expression data, since it is invariant under monotonous transformations of the data.

References

[1] Papadimitriou, C. (1995) *Computational Complexity.* (Addison-Wesley, Reading, MA).

[2] Tsafrir, D, Tsafrir, I, Ein-Dor, L, Zuk, O, Notterman, D. A, & Domany, E. (2005) *Bioinformatics* **21**, 2301–2308.

[3] Jonker, R & Volgenant, A. (1987) *Computing* **38**, 325–340.

[4] Su, A, Wiltshire, T, Batalov, S, Lapp, H, Ching, K, Block, D, Zhang, J, Soden, R, Hayakawa, M, Kreiman, G, Cooke, M, Walker, J, & Hogenesch, J. (2004) *Proc Natl Acad Sci U S A* **101**, 6062–6067.

[5] Hubbard, T, Andrews, D, Caccamo, M, Cameron, G, Chen, Y, Clamp, M, Clarke, L, Coates, G, Cox, T, Cunningham, F, *et al.* (2005) *Nucleic Acids Res.* **33**, D447–D453.

[6] R Development Core Team. (2005) *R: A Language and Environment for Statistical Computing* (R Foundation for Statistical Computing, Vienna, Austria).

[7] Gentleman, R. C, Carey, V. J, Bates, D. M, Bolstad, B, Dettling, M, Dudoit, S, Ellis, B, Gautier, L, Ge, Y, Gentry, *et al* (2004) *Genome Biology* **5**, R80.

[8] Durinck, S, Moreau, Y, Kasprzyk, A, Davis, S, De Moor, B, Brazma, A, & Huber, W. (2005) *Bioinformatics* **21**, 3439–3440.